LA-UR-02-5222

Title: A Comparison Between the Earth Simulator and AlphaServer Systems using Predictive Application Performance Models

Author(s): Darren J. Kerbyson
Adolfy Hoisie
Harvey Wasserman

Parallel Architectures and Performance Team
Modeling, Algorithms and Informatics Group (CCS-3)

Submitted to: Computer Architecture News (ACM), December 2002

16th October 2002

A Comparison Between the Earth Simulator and AlphaServer Systems using Predictive Application Performance Models

Darren J. Kerbyson, Adolfy Hoisie, Harvey J. Wasserman
Parallel Architectures and Performance Team
Modeling, Algorithms and Informatics Group, CCS-3
Los Alamos National Laboratory
Los Alamos, NM 87545
{djk,hoisie,hjw}@lanl.gov

Abstract

This work gives a detailed analysis of the relative performance between the recently installed Earth Simulator and systems built using Alpha processors. The Earth simulator is based on the NEC SX-6 vector processing node interconnected using a single-stage cross-bar network, whereas the HP AlphaServer nodes use superscalar microprocessors and are interconnected using the Quadrics fat-tree network. The performance that can be achieved results from an interplay of system characteristics, application requirements and scalability behavior. Detailed performance models are used here to predict the performance of two codes representative of ASCI computations, namely SAGE and Sweep3D. The performance models encapsulate fully the behavior of these codes and have been previously validated on many large-scale systems. They do not require access to a full sized system but rather rely on characteristics of the system as well as knowledge of the achieved single-processor performance. One result of this analysis is in the determination of an equivalent-sized Alpha-based machine that would be required to obtain the same performance as the Earth Simulator.

1. Introduction

In this work we consider a performance comparison between the Earth Simulator [8,11], and systems constructed from commodity AlphaServer nodes such as that being used in ASCI Q at Los Alamos National Laboratory (LANL) [1] and also at the Pittsburgh Supercomputing Center. The Earth Simulator was recently installed at Yokohama City, Japan. It had the design goal of giving a 1000-fold increase in the processing capability available for atmospheric research, compared to that available when envisioned in 1996, and was initially expected to achieve sustained performance of 5 Tflops (12.5% of system-peak) [16]. Current indications are that some codes are achieving a considerably higher performance, such as that currently in consideration for the Gordon Bell prize this year (with an achieved rating of over 26 Tflops per second or greater than 60% of system-peak [3]).

At present there is much interest in comparing the relative performance between the Earth Simulator and other large-scale Teraflop systems, in part due to the use of vector processors in comparison to COTS (Commodity-Off-The-Shelf) superscalar microprocessors with cache-based memory systems. However, it is a complex task to compare the performance of these systems without using simplistic metrics (such as peak flop rating). Thus, comparing the performance of the Earth Simulator with other systems has been restricted so far to a small number of applications that have actually been executed on the system, or to considering the overall system-peak performance or individual sub-system performance characteristics such as achieved MPI intra-node communication bandwidth and latencies [15]. Peak performance figures are often misleading and do not correlate in any way to the time-to-solution for a particular application.

There is a large difference in the relative processor peak performances between the two kinds of systems. An Earth Simulator node (based on an NEC SX-6) contains 8 processors each with a peak processing capability of 8 Gflops. In contrast, an AlphaServer ES45 contains 4 processors each with a peak of 2.5 Gflops. The communication network is also quite different – the Earth Simulator has a cross-bar interconnect with a 16GB/s bandwidth between any two nodes in comparison to a Quadrics QsNet fat-tree topology interconnecting ES45 nodes with a 300MB/s bandwidth. These differences affect both the surface-to-volume ratio of applications [2] along with their parallel overheads.

The peak performance of a system results from the underlying hardware architecture including processor design, memory hierarchy, inter-processor communication system, and also their interaction. But, the achievable performance is dependent upon the workload that the system is to be used for, and specifically how this workload utilizes the resources within the system.

Performance modeling is a key approach that can provide information on the expected performance of a workload given a certain architecture configuration prior

to execution. The approach that we take in this work is application centric. It involves an understanding of the processing flow in the application, the key data structures, and how they use and are mapped to the available resources. From this, a performance model is constructed that encapsulates its key performance characteristics. The aim of the model is to provide insight. By keeping the model general while not sacrificing accuracy, it may be used to explore the possible achievable performance in new situations – both in terms of hardware systems and in terms of code modifications. This approach has been successfully used on applications that are representation of the Accelerated Strategic Computing Initiative (ASCI) workload including: an adaptive mesh code [5], structured and unstructured mesh transport codes [4,7], and a Monte-Carlo particle simulation code [9].

We use two existing application performance models in this paper, one of an S_N transport application on Cartesian grids and one of a hydro code, in order to compare the relative performance between the Earth Simulator and a commodity based AlphaServer System using HP AlphaServer ES45 nodes. The models have already been validated with high accuracy on all ASCI machines constructed to date. They have also been used to validate performance during the installation of the first phase of ASCI Q [6] and in the recent procurement of ASCI Purple. The models encapsulate the performance characteristics of the processing nodes, the communication networks, the mapping of the sub-domains to processors, and the processing flow of the application along with their scaling behavior.

The performance models are used to predict the time-to-solution of the applications when considering a typical utilization of the available memory on each system. In further analysis the models are also used to "size" an AlphaServer system which would result in the same achieved performance on the two applications as the Earth Simulator.

In Section 2 an overview of both the Earth Simulator and AlphaServer ES45-based systems is given along with their salient performance characteristics. In Section 3 we describe the application characteristics and how they utilize the available resources as a function of processor count and problem size. For both applications we include an overview of their respective performance models that are utilized to provide performance predictions on large-scale systems. In Section 4, we give a comparison of the two systems both in terms of their expected time-to-solution, and also to "size" an AlphaServer system that will provide the same performance as the Earth Simulator.

2. System Comparison

The Earth Simulator and a tera-scale AlphaServer system are quite different in both their processor node architecture and inter-node connectivity. The main aspects of the system architecture of both systems are described below.

2.1 The Earth Simulator

The Earth Simulator is a distributed memory system consisting of 640 nodes inter-connected by a single stage 640x640 crossbar network which was specifically designed for this architecture. [14]. Each node is an SMP composed of eight arithmetic processors, a shared memory of 16GB, a remote access unit (RCU), and an I/O processor (IOP) as shown in Figure 1. The nodes are based on the NEC SX-6 [12] but have a reduced memory capacity and increased memory performance.

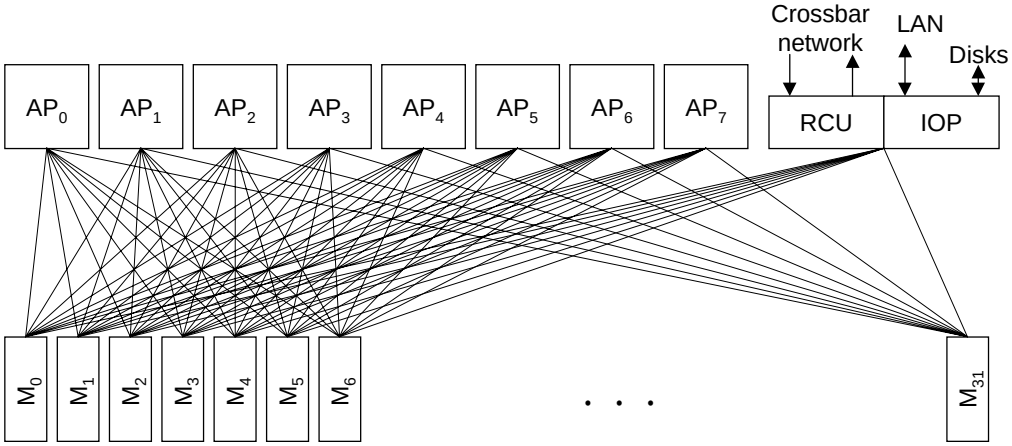


Figure 1. Internal architecture of an Earth Simulator node

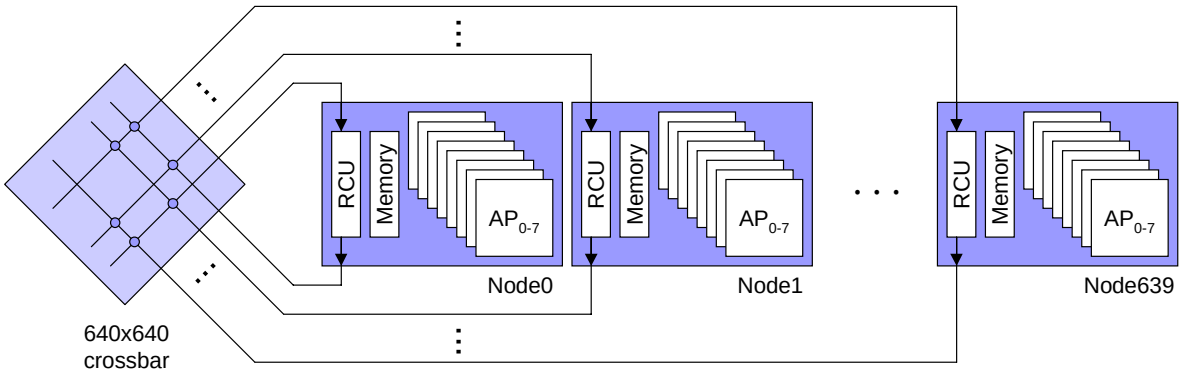


Figure 2. Crossbar interconnection between nodes in the Earth Simulator

Each arithmetic processor is connected to 32 memory units with a bandwidth of 32 GB/s (256 GB/s aggregate within a node). The peak performance of each arithmetic processor is 8 Gflops. Thus the overall system-peak performance of the Earth Simulator, based on 5120 vector processors, is 40 Tflops.

The Earth Simulator uses 0.15micron technology with air-cooling. Each arithmetic processor is contained on a single chip of an approximate size 20mm x 20mm and operates at 500 MHz. Each arithmetic processor contains a set of 8 vector units along with a 4-way super-scalar processor. Each vector unit has six types of operation: add/shift, multiply, divide, logical, mask, and load/store. There are 72 vector registers of 256 vector elements. The scalar processor has 64KB Instruction and Data caches, along with 1KB of general-purpose scalar registers.

The RCU connects a node to the crossbar switch as shown in Figure 2. The inter-node communication has a peak bandwidth of 16GB/s in each direction based on bi-directional communication. Each communication channel is 128bits wide in each direction with a peak transfer rate of 1Gbits per second. The minimum latency for an MPI level communication is quoted as 5.2µsec within a node, and 8.6µsec between nodes [8].

2.2 The AlphaServer ES45 Supercomputer

The AlphaServer ES45 can be purchased as a single node or in large multiples to create a large-scale system. It is usually used in conjunction with the Quadrics QsNet fat-tree communication network [10] to deliver tera-scale performance at computing facilities such as at the Pittsburgh Supercomputing Centre (PSC), and for ASCI Q at Los Alamos National Laboratory (LANL) [1]. Each ES45 contains four 21264D EV68 Alpha microprocessors which can deliver a peak of 2.5Gflops with its latest operating frequency of 1.25GHz. The largest two installations at the time of writing consist of 768 nodes (PSC) and 1024 nodes (part of ASCI Q at LANL). The system at PSC currently utilizes the 1GHz Alpha processors and the system at LANL utilizes the faster 1.25GHz processors giving a current system-peak performance of 6Tflops and 10Tflops respectively. ASCI Q is expected to grow to a system giving 30Tflops system-peak performance.

Each 1.25GHz EV68 Alpha processor contains 64KB L1 instruction and data caches, and a 16MB unified L2 cache. Each ES45 node is a 4-way SMP containing a maximum of 32GB of memory. A peak memory bandwidth up to 8GB/s is possible using two 256-bit memory buses running at 125MHz. Four independent 64-bit PCI buses (running at 66 MHz) provide I/O capabilities. The internal architecture of an ES45 node is shown in Figure 3.

ES45 nodes are interconnected using the Quadrics QsNet as shown in Figure 4. Each node contains one or two Elan PCI communication cards. These connect to a multi-stage Elite 4x2 switching element to implement a quaternary fat tree. A dimension k quaternary fat-tree consists of k levels each with 4^{k-1} Elite switches for a 4^k node system. The peak bandwidth achievable on an MPI level communication is 300MB/s with a typical latency of 5µsec. The latency increases slightly with the physical distance between nodes. A detailed description of the Quadrics network can be found in [10].

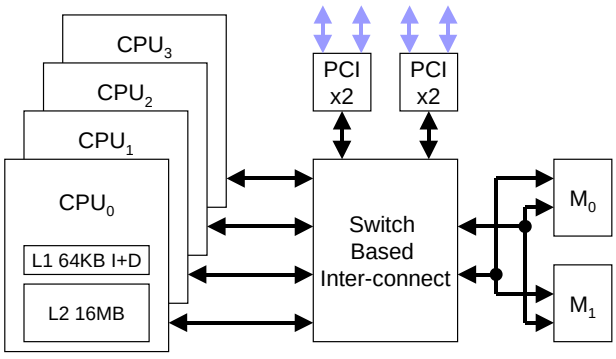


Figure 3. ES45 Internal architecture

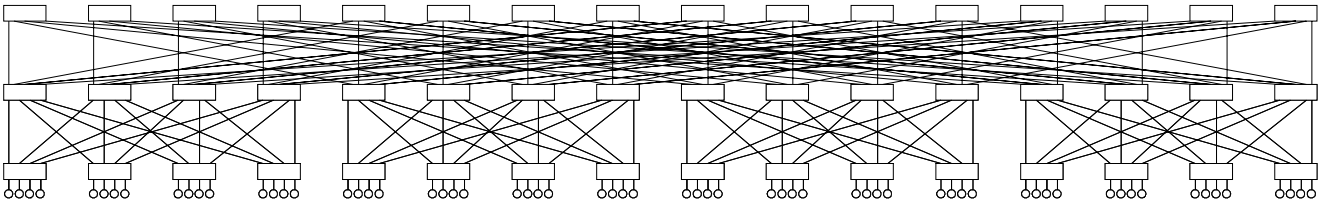


Figure 4. Topology for a dimension 3 quaternary fat-tree network for a 64 node ES45 system

2.3 Comparison overview of the two systems

It is clear that the Earth Simulator has more powerful processors, with a peak performance of 8Gflops (vs. 2.5Gflops), but fewer processors than that expected in the larger AlphaServer Systems (such as ASCI Q). However, the amount of memory per processor is only 2GB on the Earth Simulator vs. up to 8 GB on the Alphas. The main memory bandwidth is also higher on the Earth Simulator (256GB/s vs. 8GB/s) although the two levels of cache on the Alpha microprocessors help increase the effective memory bandwidth. The inter-node communication latency is lower on the Alpha system, 5µs vs. 8.6µs. The achievable communication bandwidth on the Earth Simulator is a factor of 29 higher (11.8GB/s vs. 300MB/s). The main characteristics of the two systems are summarized in Table 1.

Note: the inter-node MPI communication performance for both systems is based on measured performance already reported for unidirectional inter-node communication [13, 10].

It should also be noted that the largest problems that can be processed on a single processor of the Earth Simulator are a factor of 2 smaller than that on an ES45 processor, given the memory currently available on the 2 machines.

Table 1. Characteristics of the Earth Simulator and AlphaServer ES45 systems

	Earth Simulator (based on NEC SX-6)	AlphaServer (HP ES45)
Node Architecture	Vector SMP	Microprocessor SMP
System Topology	Crossbar (single-stage)	Fat-tree
Number of nodes	640 (maximum)	1+ (3072 expected ASCI Q)
Processors - per node	8	4
- system total	5120	(12288 expected ASCI Q)
Processor Speed	500 MHz	1.25 GHz
Peak speed - per processor	8 Gflops	2.5 Gflops
- per node	64 Gflops	10 Gflops
Memory - per node	16 GB	16 GB (max 32 GB)
- per processor	2 GB	4 GB (max 8 GB)
- system total	10.24 TB	(~48 TB expected ASCI Q)
Memory Bandwidth (peak)		
- L1 Cache	N/A	20 GB/s
- L2 Cache	N/A	13 GB/s
- Main memory (per processor)	32 GB/s	2 GB/s
Inter-node MPI communication		
- Latency	8.6 µsec	5 µsec
- Bandwidth	11.8 GB/s	300 MB/s

3. Representative ASCI Applications

The performance of the Earth Simulator is compared to that of an AlphaServer system by using two full applications representative of the ASCI workload, namely SAGE and Sweep3D. In this analysis, measurement of the applications is not currently possible due to the limitation in access to the Earth Simulator, and also due to the largest Alpha system currently limited to 1024 nodes. Rather in this analysis we use detailed performance models of each of the two codes. These models have high accuracy, as shown through a validation on all existing ASCI system. A description of both SAGE and Sweep3D is included below. Details on the performance model of each code have been previously described [4,5], and an overview is included in the appendices.

The performance of an application results from many factors and not just from the peak processing capability of a system. The achieved performance reflects the way in which the application used the available resources within a system. For instance, consider the partitioning of a 3-D data grid in one, two and three dimensions as shown in Figures 5(a)-(c) respectively. In each figure, the data exchanges performed on a gather (or scatter) are shown for one central sub-grid of the data. In the 2-dimensional partitioning 5(b), boundary data is exchanged with each of its four neighbors.

Various data decompositions lead to different surface-to-volume properties of the application [2]. For example, the communication requirements generated by the 1-D decomposition depicted in Figure 5(a) are going to be bandwidth dominated, while those generated by the 3-D decomposition of Figure 5(c) are likely to be latency bound. Moreover, the scalability properties of these decompositions are different. If the codes are to be run in a weak scaling fashion, as they are on the ASCI machines, 1-D data decomposition will not be scalable, the communication requirements will increase with the processor count even as the sub-grid size remains constant. The 3-D data decomposition on the other hand will maintain a constant surface-to-volume in the weak scaling scenario.

3.1 SAGE

SAGE (SAIC's Adaptive Grid Eulerian hydrocode) is a multidimensional (1D, 2D, and 3D), multi-material, Eulerian hydrodynamics code with adaptive mesh refinement (AMR). It comes from the Los Alamos National Laboratory's Crestone project, whose goal is the investigation of continuous adaptive Eulerian techniques to stockpile stewardship problems. SAGE represents a large class of production ASCI applications at Los Alamos that routinely run on 1,000s of processors for months at a time.

Adaptive mesh refinement operations are performed on cells as necessary at the end of each processing cycle. Each cell at the topmost level (level 0) can be considered as the root node of an oct-tree of cells in lower levels. For example, the shock-wave indicated in the 3-D spatial domain in Figure 6 by the solid line may cause cells close to it to be split into smaller cells. In this example, a cell at level 0 is not refined, while a cell at level n is a domain 8^n times smaller.

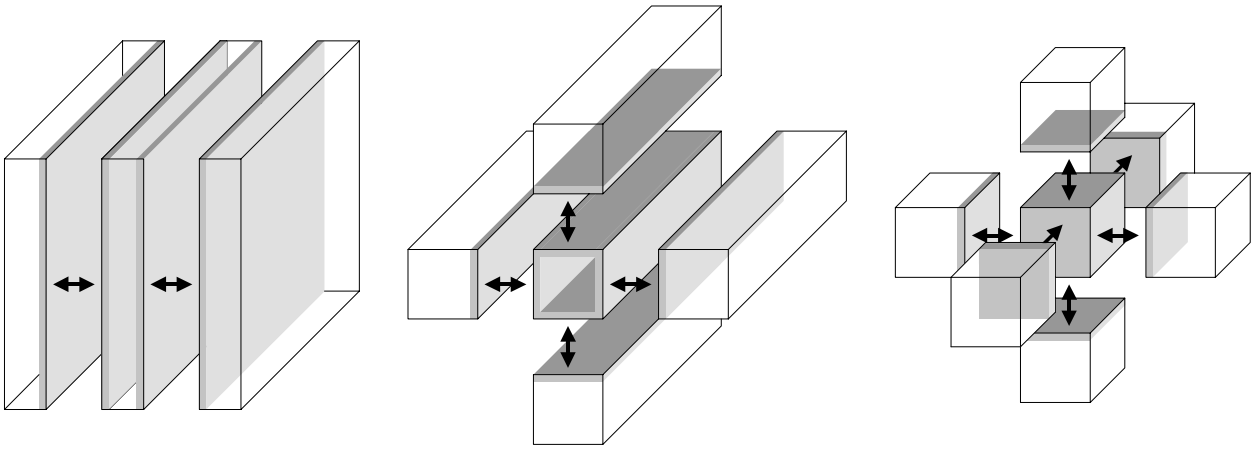


Figure 5. Data Decomposition of a 3-D data grid, showing boundary exchanges on a gather (or scatter) operation, in (a) 1 dimension, (b) 2 dimensions, and (c) 3 dimensions

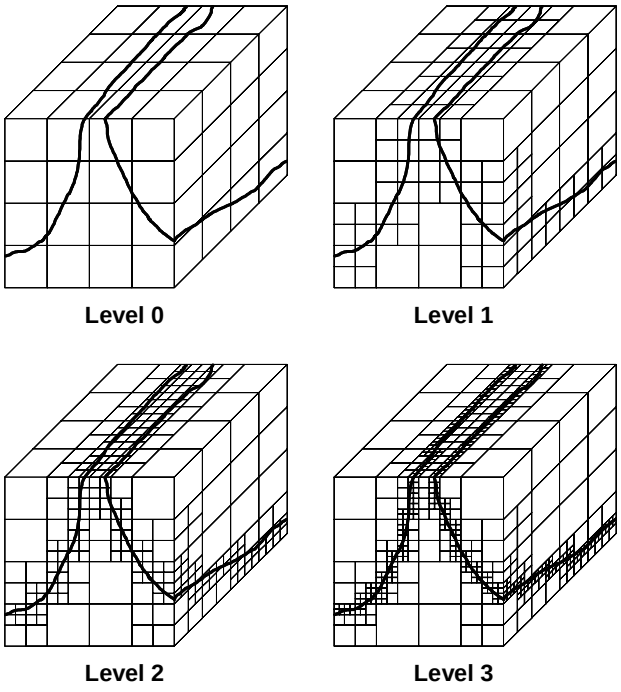


Figure 6. Example of Adaptive Mesh Refinement at multiple levels

The key characteristics of SAGE are:

Data decomposition – The spatial domain is partitioned across processors in 1-D *slab* sub-grids. The problem size grows proportionally with the number of processors in the normal operational mode of SAGE, i.e. a weak-scaling characteristic.

Processing flow – the processing proceeds in cycles. In each cycle there are a number of stages that involve the three operations of: one (or more) data gathers to obtain a copy of remote neighbor data, computation on each of the gathered cells, and one (or more) scatter operations to update data on remote processors.

AMR and load-balancing – at the end of each cycle, each cell can either be split into a block of smaller 2x2x2 cells, combined with its neighbors to form a single larger cell, or remain unchanged. A load-balancing operation takes place if any processor contains 10% more cells than the average cells across all processors.

The 1-D decomposition leads to a number of important factors that influence the achievable performance. For instance the amount of data in gather-scatter communication increases, and also the distance between processors increases as the number of processors increases. The performance model of SAGE is included in Appendix A. Full details on the characteristic scaling behavior of SAGE are given in [5].

3.2 Sweep3D

Sweep3D is a compact application taken from the DOE Accelerated Strategic Computing Initiative (ASCI) workload. It is a time-independent, Cartesian-grid, single-group, "discrete ordinates" deterministic particle transport code. Estimates are that deterministic particle transport accounts for 50-80% of the execution time of many realistic simulations on current DOE systems; this percentage may expand on future 100-Tflops systems. The basis for neutron transport simulation is the time-independent, multigroup, inhomogeneous Boltzmann transport equation [4].

Sweep3D is characteristic of a larger class of algorithms known as wavefront algorithms. These algorithms exhibit a complex interplay between their surface-to-volume and processor utilization. As such the performance prediction of Sweep3D cannot be done without a detailed application model. A brief description of the model developed at Los Alamos is given in Appendix B.

The 3D spatial domain in Sweep3D is mapped to a logically 2D processor array. Wavefronts (or "sweeps") scan the processor array originating in all corners in a pipelined fashion as shown in Figure 7. The bigger the subgrid size, the more favorable the surface-to-volume is, but the processor utilization decreases. In order to achieve optimality between the two, the code uses blocking in one spatial dimension and in angles. The tuning of the blocking parameters has an important effect on the runtime of the application. Our model captures the effect of the blocking parameters, hence allowing the selection of those values that minimize the runtime.

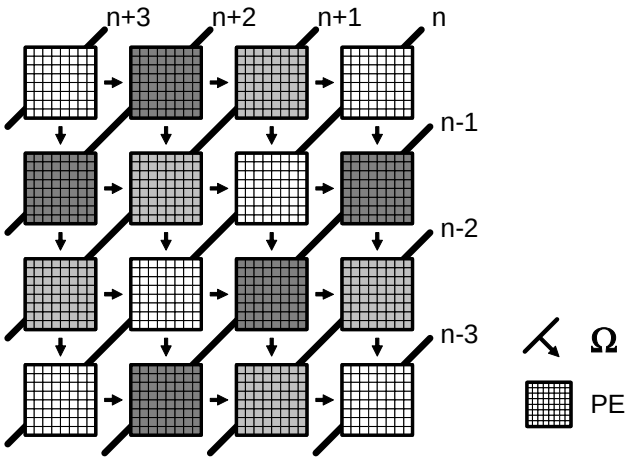


Figure 7. The pipeline processing of sweeps on a 2D processor array. n denotes a sweep that is currently processed on the major diagonal of the processor array. Other wavefronts shown (n-3…n+3) are either ahead or behind sweep n.

4. Predictive Comparison

In this section we describe the performance that will be achieved on the Earth Simulator by the two codes representative of the ASCI workload – SAGE and Sweep3D. The types of computations that these codes encapsulate represent a high percentage of the cycles that are used on the ASCI machines. The performance estimates are predicted here strictly based on our validated models without any direct measurement of SAGE or Sweep3D on either an NEC SX-6 or the Earth Simulator itself. However, they take into account all architectural details that are parameters to our models as well as low-level performance data found in the Earth Simulator literature, such as the latency and bandwidth for MPI communications [8]. In order to produce the estimate of the runtime of these applications on the Earth Simulator we have used the performance models as discussed in Section 3 and detailed in the appendices.

In this comparison we use both SAGE and Sweep3D in a weak-scaling mode in which the sub-grid sizes on each processor remain a constant. However, since the main memory per processor on the Earth Simulator is a factor of 2 less than that in an Alpha ES45 we use sub-grid sizes that are also in this ratio. This corresponds to the applications using all the available memory. Both the weak scaling scenario and the use of as much memory as possible are typical of they way in which large-scale ASCI computations are performed.

Both performance models are a function of the time to compute a subgrid of data on a single processor. Currently the percentage of peak floating point speed that will be achieved by SAGE and Sweep3D on a single processor of the Earth Simulator is unknown. To circumvent this, in this analysis, we predict the runtime of the applications on a family of curves. Each curve assumes a speed of either: 5%, 10%, 15%, 20%, 25%, or 30% of single processor-peak. It will be possible to better quantify this value when measurements are obtained from an NEC SX-6. It should be noted that SAGE currently achieves approximately 10% of single processor-peak performance on microprocessor systems such as the AlphaServer ES45 used in ASCI Q, and Sweep3D achieves approximately 14%. Both codes may need to be modified (at worst re-coded) in order to take advantage of the vector processors in the Earth Simulator. However, none of these codes is particularly tuned for the architectural features of the RISC architectures, particularly the on-chip parallelism and the memory hierarchy.

Further parameters which have not been measured are related to the inter-node communication performance. This would require access to the cross-bar network of the Earth Simulator for a higher accuracy. Several architectural aspects of the communication subsystems have been taken into account in order to calculate the best estimate for these parameters. Specific assumptions relate to the serialization of messages when several processors within a node perform simultaneous communications to another node on the Earth Simulator. This results in a multiplicative factor on the time taken to perform a single communication (which is true for the Quadrics network). The analysis below is sensitive to this parameter. The values used for all the parameters used in the performance models are listed in Appendix A, and Appendix B. In addition, uniform bandwidth and latency is assumed between any two nodes in both systems.

4.1 SAGE - Earth Simulator vs. AlphaServer ES45 performance

We compare in Figure 8 the performance of the Earth Simulator with that of the AlphaServer ES45 (1.25-GHz) for SAGE. Performance is compared for various processor counts on each system. However, since each Earth Simulator processor has an 8Gflops peak performance, and each ES45 Alpha processor has a 2.5Gflops peak performance, the relative advantage of the Earth Simulator based on peak performance alone is a factor of 3.2 (indicated by a single horizontal line).

A value greater than 3.2 in Figure 8 indicates a performance advantage of the Earth Simulator compared to the Alpha system over and above their ratio of single-processor peak performance. This analysis assumes a typical problem size for SAGE of 35,000 cells per processor on an Alpha, and 17,500 cells per processor on the Earth Simulator. The difference in the number of cells per processor between the two systems reflects the factor of two in the main memory sizes.

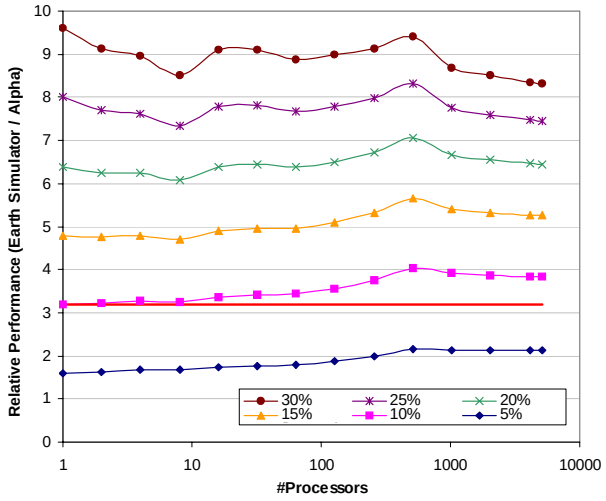


Figure 8. SAGE Performance comparison between the Earth Simulator and an Alpha ES45 system

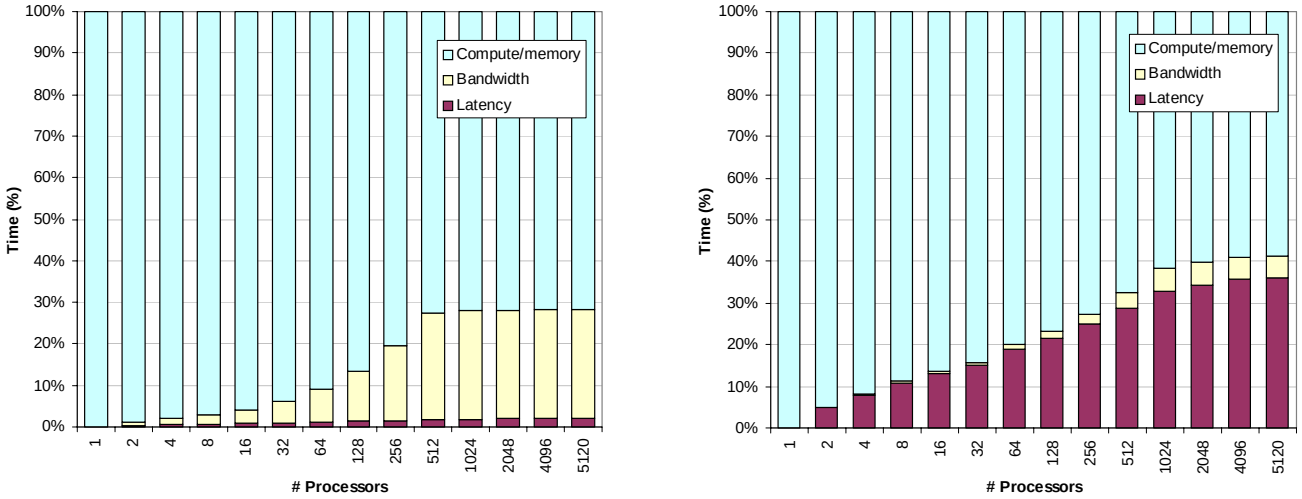


Figure 9. Time spent in processing and communication in SAGE on (a) an Alpha ES45 system and (b) the Earth Simulator

It can be seen that the relative performance is better on the Earth Simulator in all cases. Moreover, if performance greater than 10% of single processor-peak is assumed, the performance advantage of the Earth Simulator becomes larger than just the ratio of the single-processor peak of 3.2. Depending on the percentage of single processor-peak that will be achieved, the performance advantage of the Earth simulator on the largest configuration is between a factor of 2 and a factor of 8.

The shape of the curves in Figure 8 indicates the various surface-to-volume regimes as the machine sizes increase. Clearly the y-axis intercepts for all curves indicate the difference in processing speed for each assumed percentage of single processor-peak achieved on the Earth Simulator. As the machine size increases, the communication starts having a bearing on the surface-to-volume. In particular, for very large configurations, for which the communication requirements become very large, the bandwidth plays an important role in the performance advantage of the Earth Simulator.

The difference in the make up of the achieved performance of both systems can be seen in Figure 9(a) and Figure 9(b). For each system, the percentage of time spent either processing (compute/memory), or in communication (latency or bandwidth) is shown. It can be seen that on larger processor counts, a greater percentage of time is taken by the bandwidth component on the Alpha, in comparison, the Earth Simulator is not subject to bandwidth costs but rather is more subject to latency costs, simply due to its much larger available bandwidth.

4.2 SAGE - Equivalent Node Count for the Earth Simulator and AlphaServer ES45

Here, we consider what size of AlphaServer ES45 (1.25GHz) system would achieve the same performance as the Earth Simulator. In this comparison we again assume a typical problem size of SAGE of 35,000 cells per processor on the Alpha system, and 17,500 cells per processor on the Earth simulator. Of course this normalization is strictly done for the purpose of factoring the memory advantage of the Alphas, and does not equate the processing time.

Figure 10 shows the estimated number of Alpha ES45 nodes required in order to achieve the same performance as the Earth Simulator on SAGE. Note that one node has a peak performance of 10 Gflops (hence 1,000 have a peak of ~10Tflops). The bold numbers above each bar indicate the size of the equivalent Alpha system in Tflops. If 10% of single processor-peak performance is obtained on SAGE on the Earth Simulator, then 4,900 Alpha nodes will be required to obtain the same performance. Thus a 49Tflop-peak Alpha system would be required to achieve the same performance as the 40Tflop-peak Earth Simulator. For comparison, the 30Tflop ASCI Q is expected to contain 3,072 nodes. The size of an AlphaServer equivalent to the Earth Simulator grows with the increase of the achieved single processor speed on the Earth Simulator.

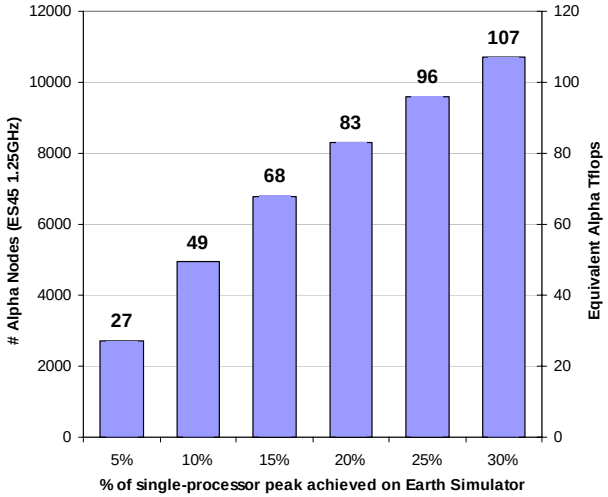


Figure 10. Number of Alpha ES45 nodes equivalent to the Earth Simulator (on SAGE)

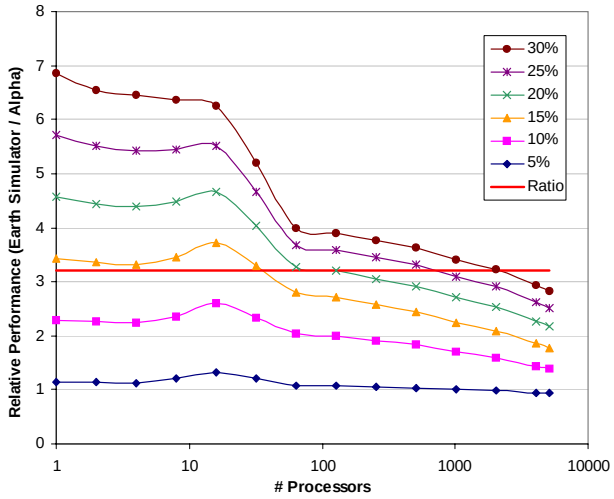
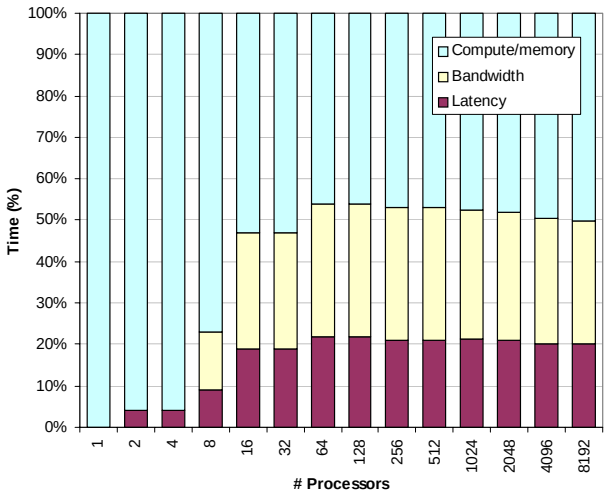


Figure 11. Sweep3D Performance comparison between the Earth Simulator and an Alpha ES45 system

4.3 Sweep3D - Earth Simulator vs. AlphaServer ES45 performance

We compare in Figure 11 the performance of the Earth Simulator with that of the AlphaServer ES45 (1.25-GHz) for Sweep3D. Performance is compared using the same basis to that for SAGE. A sub-domain size in Sweep3D of 12x12x280 (40320 cells) is used per processor on the Alpha and an 8x8x280 (17920 cells) sub-domain is used on the Earth Simulator.

It can be seen that the relative performance on the Earth Simulator decreases with the increase in the number of processors used. Depending on the percentage of single-processor peak that will be achieved, the performance advantage of the Earth simulator on the largest configuration is between a factor 1 and 3. The Earth Simulator performs worse than the relative single processor-peak speed advantage of 3.2 on the largest configurations considered. This is due in part to the communication requirements in this application being largely latency-bound, and also in part due to the difference in the scaling behavior of the different problem sizes as a result of the difference in memory capacities. Hence the lower ratios in these regimes compared to SAGE. The importance of latency can be seen in the comparison between Figure 12(a) and 12(b). The Earth simulator has a much larger latency component than on the Alpha system.



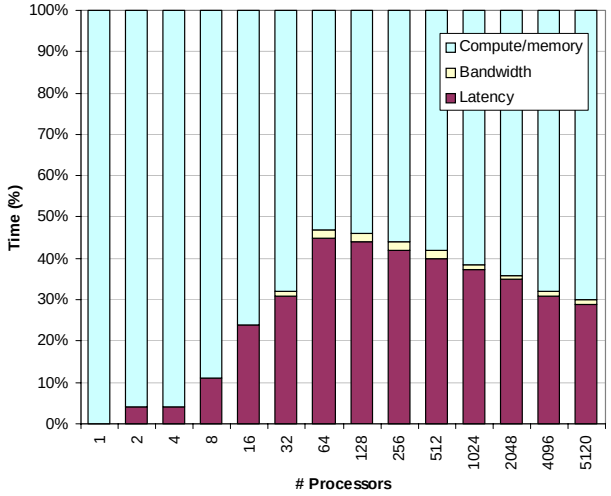


Figure 12. Time spent in processing and communication in Sweep3D on (a) an Alpha ES45 system and (b) the Earth Simulator

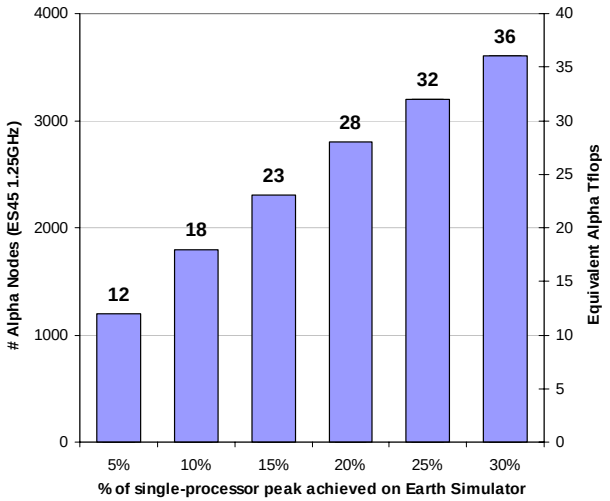


Figure 13. Number of Alpha ES45 nodes equivalent to the Earth Simulator (on Sweep3D)

4.4 Sweep3D - Equivalent Node Count for the Earth Simulator and Alpha ES45.

The equivalent number of Alpha ES45 (1.25 GHz) nodes that are required in order to achieve the same performance as the Earth Simulator for Sweep3D is shown in Figure 13. If 15% of single-processor peak performance is obtained on Sweep3D on the Earth Simulator, then 1,800 Alpha nodes will be required to obtain the same performance. Again, compare with a 30Tflop-peak ASCI Q which is expected to contain 3,072 nodes.

4.5 Composite Workload Comparison

In this comparison we assume a hypothetical workload consisting of 40% SAGE and 60% Sweep3D. Table 2 shows the equivalent sized Alpha system (in terms of system-peak Tflops) to the Earth Simulator using this workload profile. Table 2 includes the same levels of assumed achieved percentage of single-processor peak as above for both applications on the Earth Simulator.

It can be seen in Table 2 that if both of the codes were to achieve a high percentage of single-processor peak on the Earth Simulator (e.g. 25% of peak on SAGE and 25% of peak on Sweep3D), the peak rating of an equivalent Alpha system would be 57.6Tflops (44% higher than that of the Earth Simulator). However, if a lower percentage of single-processor peak is achieved on both codes then the size of the Alpha system could be considerably less. The information in Table 2 could be coupled with the cost of the two systems to determine which system provides better price-performance for a given cost.

Table 2. Peak-Tflop rated Alpha system required to achieve the same performance of the Earth Simulator assuming application weighting (SAGE 40%, Sweep3D 60%). (Numbers in this table represent peak performance in Tflops)

		SAGE % of single-processor peak					
		5%	10%	15%	20%	25%	30%
Sweep3D	5%	18.0	26.8	34.4	40.4	45.6	50.0
	10%	21.6	30.4	38.0	44.0	49.2	53.4
	15%	24.6	33.4	41.0	47.0	52.2	56.6
	20%	27.6	36.4	44.0	50.0	55.2	59.6
	25%	30.0	38.8	46.4	52.4	57.6	62.0
	30%	32.4	41.2	48.8	54.8	60.0	64.4

At present we expect Sweep3D to achieve a low percentage of single-processor peak on the Earth Simulator. This educated guess is based on the fact that Sweep3D vectorizes poorly in its current implementation. We anticipate that SAGE will achieve a higher percentage of peak performance on the Earth Simulator, mainly due to the fact that its solver could vectorize relatively well, and the profiling of the code shows that the solver takes a considerable chunk of its runtime. It is important to note however, that these codes may be optimized over time for better vectorization, leading to higher achieved single processor performance on the Earth Simulator. Thus the information presented here for a family of curves, with each curve being based on a different level of achieved single processor performance, provides information that will remain valid over time, over lasting increasing optimization of the codes. We re-state here that the current implementations of these codes is not optimized for RISC architectures either, hence the comparison between the two machines using the current implementation is fair.

5. Conclusions

We have compared the performance of the Earth Simulator against that of an Alpha ES45-based system for two applications representative of the ASCI workload. The applications considered are Sweep3D, representative of S_N transport computations and SAGE, representative of hydro computations.

All performance data for the applications were generated using highly accurate models for the runtime of these applications developed at Los Alamos and validated on a variety of large-scale parallel systems, including all

ASCI machines. In order to bypass the limitations of not having had the opportunity to run these applications on the Earth Simulator to obtain single-processor performance, a family of curves was generated for the Earth Simulator that assume a performance of 5, 10, 15, 20, 25 and 30% of single-processor peak speed.

We have analyzed the performance for the two machines as they are. No architectural changes of any kind were considered. That includes the amount of memory with which these systems are equipped, the Earth Simulator having 2GB/processor and the Alpha system having 4GB/processor. In this way we have tried to avoid the dangers of analyzing a moving target, choosing instead to compare snapshots of the machines in their current configuration.

Our analysis shows that for all assumed serial performance of the 2 applications on the Earth Simulator, there is a performance advantage of this machine over the Alpha system on an equivalent processor count. The advantage is more pronounced for SAGE, in which computation and bandwidth are the main contributors to the performance of the code. For SAGE the performance on the Earth Simulator is between a factor of 2-8 larger than on the Alpha system on an equivalent processor count. By the same metric, on Sweep3D, while the Earth Simulator maintains the performance advantage, the relative gain is only between a factor of 1-3.

By considering a number of values of the achieved single-processor performance on the Earth Simulator we think that we also covered the grounds given the distinct possibility that no single value will be generated by various benchmarkers. A multitude of variants of these codes may exist, some of which may vectorize better than others.

An intuitive, but unsubstantiated by a thorough analysis, principle was submitted to the scientific community a decade ago. The argument went as follows: specialized processors (including vector processors) are faster but expensive as they are not backed up by the marketplace, commodity production. Hence, since price-performance is the most important consideration, let's compensate the performance disadvantage of the COTS microprocessors by building large-scale machines with a greater number of them.

In this paper we quantify this thesis, by showing how large an Alpha-based system would be required in order to achieve an equivalent performance as the Earth Simulator for the ASCI workload under consideration. One direct conclusion is that peak performance is a poor indicator of real performance, and hence price-performance based on peak is also a poor indicator. For example, assuming a sustained single-processor performance for Sweep3D at between 5-10% of peak performance and 20-25% of peak for SAGE, and a workload consisting of 60% Sweep3D and 40% SAGE, the equivalent Alpha-based system that would achieve the same sustained performance as the Earth Simulator would need to have a peak performance of between approximately 40-49 Tflops.

This work clearly shows that only through modeling such analysis can be performed given the multi-dimensional space of performance that needs to be considered. This performance space is highly complex and non-linear for large-scale parallel applications and architectures.

Acknowledgements

The authors would like to thank Scott Pakin, Fabrizio Petrini and Kei Davis of Los Alamos for their comments on this work. This work was funded by ASCI Institutes.

Los Alamos National Laboratory is operated by the University of California for the National Nuclear Security Administration of the US Department of Energy.

References

[1] ASCI Q at Los Alamos, http://www.lanl.gov/asci

[2] S. Goedecker, A. Hoisie, Performance Optimization of Numerically Intensive Codes, SIAM Press, ISBN 0-89871-484-2, March 2001.

[3] Gordon Bell Award, SC2002, Baltimore, http://www.sc2002.org

[4] A. Hoisie, O. Lubeck, H. Wasserman, "Performance and Scalability Analysis of Teraflop-Scale Parallel Architectures using Multidimensional Wavefront Applications", *Int. J. of High Performance Computing Applications*, Vol. 14, No. 4, Winter 2000, pp. 330-346.

[5] D.J. Kerbyson, H.J. Alme, A. Hoisie, F. Petrini, H.J. Wasserman, M. Gittings, "Predictive Performance and Scalability Modeling of a Large-Scale Application", in Proc. SC2001, Denver, November 2001.

[6] D.J. Kerbyson, A. Hoisie, H.J. Wasserman, "Use of Predictive Performance Modeling During Large-Scale System Installation", in Proc. of 1st Int. Workshop on Hardware/Software support for Parallel and Distributed Scientific and Engineering Computing, PACT-SPDSEC02, Charlottesville, September 2002.

[7] D.J. Kerbyson, S.D. Pautz, A. Hoisie, "Predictive Modeling of Parallel Sn Sweeps on Unstructured Meshes", Los Alamos National Laboratory report LA-UR-02-2662, May 2002.

[8] S. Kitawaki, M. Yokokawa, "Earth Simulator Running", Int. Supercomputing Conference, Heidelberg, June 2002.

[9] M. Mathis, D.J. Kerbyson, "Performance Modeling of MCNP on Large-Scale Systems", Los Alamos Computer Science Institute Symposium, Santa Fe, October 2002.

[10] F. Petrini, W.C. Feng, A. Hoisie, S. Coll, E. Frachtenberg, "The Quadrics Network: High-Performance Clustering Technology", *IEEE Micro*, 22(1), 2002, pp. 46-57.

[11] T. Sato, "Can the Earth Simulator Change the Way Humans Think?", Keynote address, Int. Conf. Supercomputing, New York, June 2002.

[12] The NEC SX-6, NEC product description, NEC Corporation, http://www.sw.nec.co.jp/hpc/sx-e

[13] H. Uehara, M. Tamura, M. Yokokawa, "An MPI Benchmark Program Library and Its Application to the Earth Simulator, ISHPC 2002, LNCS Vol. 2327, Springer-Verlag, 2002, pp 219-230.

[14] T. Watanabe, "A New Era in HPC: Single Chip Vector Processing and Beyond", Proc. NEC Users Group meeting, XIV, May 2002.

[15] P. Worley, "Preliminary SX-6 Evaluation', http://www.csm.ornl.gov/evaluation/sx6

[16] M. Yokokawa, "Present Status of Development of the Earth Simulator', in IWIA'01, IEEE Computer Press, 2001, pp. 93-99.

Appendix A: SAGE model

The complete model is described below. Details on the development of the model can be found in [1].

The model assumes weak-scaling - that is the sub-domain on each processor is constant for all processor counts. In SAGE, a 3-D spatial domain is assumed which is sub-divided in 1-D only. The volume of this spatial domain is:

$$V = E.P = L^3 \tag{A.1}$$

where P is the number of PEs, and E is the number of level 0 cells per PE.

The runtime for one cycle of the code is given by:

$$\begin{aligned} T_{cycle_i}(P, E, \mathbf{D}, \mathbf{A}, \mathbf{M_{cm}}) = \; & T_{comp}(E.D_i) + T_{memcon}(P, E.D_i) + \\ & T_{allreduce}(P) + T_{GScomm}(P, E, D_i) + \\ & T_{divide}(A_i) + T_{combine}(E.D_i) + T_{load}(M_{cm_i}, P) \end{aligned} \tag{A.2}$$

where **D** is the cell division factor [1..8^{maxlevel}], **A** is the maximum number of cells added (over all processors) through the AMR division process, and $\mathbf{M_{cm}}$ is the maximum number of cells moved between any two PEs in the load balancing. Note that **D**, **A**, and $\mathbf{M_{cm}}$ are defined as a vector whose elements are indexed by cycle.

$T_{comp}(E.D_i)$	is the sequential computation time for $E.D_i$ cells (normally measured).
$T_{memcon}(P,E.\ D_i)$	is the memory contention that may occur between PEs within an SMP node
$T_{GScomm}(P,E,D_i)$	is the gather and scatter communication time
$T_{allreduce}(P)$	is the allreduce communication time
$T_{divide}(A_i)$	is the time to divide cells in cycle *i*
$T_{combine}(E.D_i)$	is the time to combine cells in cycle *i*
$T_{load}(M_{cmi}, P)$	is the time to perform the load-balancing

The gather-scatter communication time is given by:

$$\begin{aligned} T_{GScomm}(P, E, D_i) = C(P, E) \Bigg(& \begin{pmatrix} f_{GS\_r}.T_{comm}(Surface_Z.MPI_{Real8}, P) + \\ f_{GS\_i}.T_{comm}(Surface_Z.MPI_{INT}, P) \end{pmatrix} + \\ & \begin{pmatrix} f_{GS\_r}.T_{comm}(Surface_Y.MPI_{Real8}, P) + \\ f_{GS\_i}.T_{comm}(Surface_Y.MPI_{INT}, P) + \\ f_{GS\_r}.T_{comm}(Surface_X.MPI_{Real8}, P) + \\ f_{GS\_i}.T_{comm}(Surface_X.MPI_{INT}, P) \end{pmatrix} \end{aligned} \tag{A.3}$$

where f_{GS_r} and f_{GS_I} are the frequency of real and integer gather-scatters per cycle (measured at 160 and 17, respectively). $Surface_Z$, $Surface_Y$, $Surface_X$ are processor bounday sizes (in words) - for the 1-D slab decomposition

$Surface_Z$ = MIN($(L.D_l)^2$, $E.D_i/2$), $Surface_Y = 2.L.D_i$, and $surface_x = 4.D_i$ words. MPI_{real8} and MPI_{INT} are determined by the MPI implementation

The contention on the processor network when using P processors, $C(P,E,)$ is given by:

$$C(P,E) = MIN\left(MAX\left(\frac{1}{CL} \frac{L^2}{surface_Z}, 1 \right), \frac{P_{SMP}}{CL} \right) \quad \text{(A.4)}$$

where CL is the number of communication links per node, and P_{SMP} is the number of PEs per node. $T_{comm}(S,P)$ is the time taken to communicate a message of size S when using P processors:

$$T_{comm}(S,P) = L_c(S,P) + \frac{S}{B_c(S,P)} \quad \text{(A.5)}$$

where L_c is the Latency and B_c is the Bandwidth of the communication network whose values vary with the message size and processor count.

The time taken to perform the all-reduce operations is modeled as:

$$T_{allreduce}(P) = f_{allred}.2.\log_2(P).T_{comm}(4,P) \quad \text{(A.6)}$$

where f_{allred} is the frequency of all-reduce operations per cycle. The memory contention is modeled as:

$$T_{memcon}(P,E.D_i) = E.D_i T_{mem}(P) \quad \text{(A.7)}$$

where $T_{mem}(P)$ is the measured memory contention on P processors per cell per cycle. The time taken to perform the cell division and cell combination at the end of each cycle is modeled as: $T_{divide}(A_i) = A_i.T_{div}$, and $T_{combine}(E.D_i) = E.D_i.T_{comb}$ respectively. T_{div} is the time to divide a single cell, and T_{comb} is the time to check and combine cells (both are measured on a single processor). The time to perform the load balancing is modeled as:

$$T_{load}(M_{cm_i},P) = N_{vars\_i}.T_{com}(M_{cm_i}.MPI_{INT},P) + N_{vars\_r}.T_{com}(M_{cm_i}.MPI_{Real8},P) \quad \text{(A.8)}$$

where N_{vars_i} , and N_{vars_r} are the number of integer and real variables that are communicated in the load balance operation.

The values of the main parameters used by the SAGE performance model in the analysis in Section 4 are listed in Table A.1. The values for the Alpha system have been previously measured. The values for the Earth Simulator are assumed using the best information currently available. These may be refined once accurate performance information is known such as the actual computation time for 17,500 cells on a single Earth Simulator processor.

Table A.1. Main parameters used in the SAGE performance model.

Parameter		Alpha ES45	Earth Simulator
P_{SMP}	Processors per node	4	8
CL	Communication Links per Node	1	1
E	Number of level 0 cells	35,000	17,500
f_{GS_r} f_{GS_I}	Frequency of real and integer gather-scatters per cycle	177 22	177 22
$T_{comp}(E)$	Sequential Computation time for E cells	$68.6\mu s$	$42.9\mu s$ (%5 *of peak*) $21.4\mu s$ (%10 *of peak*) $14.3\mu s$ (%15 *of peak*) $10.7\mu s$ (%20 *of peak*) $8.6\mu s$ (%25 *of peak*) $7.1\mu s$ (%30 *of peak*)
$L_c(S,P)$	Bi-directional MPI communication Latency	$6.10\mu s$ $S < 64$ $6.44\mu s$ $64 \le S \le 512$ $13.8\mu s$ $S > 512$	$8\mu s$
$B_c(S,P)$	Bi-directional MPI communication Bandwidth (per direction)	0.0 $S < 64$ $78MB/s$ $64 \le S \le 512$ $120MB/s$ $S > 512$	$10GB/s$
$T_{memcon}(P)$	Memory contention per cell within a node	$1.8\mu s$ $P = 2$ $4.8\mu s$ $P > 2$	0

Appendix B - Predictive Performance Model of Sweep3D

The performance model of Sweep3D has been validated on many large-scale systems including the ASCI machines. The complete model is described below for SMP based systems. Details on the development of the model can be found in [4].

The model assumes weak-scaling - that is the sub-domain on each processor is constant for all processor counts. The spatial domain is sub-divided in 2D. The overall size is $N_x.P_x$ by $N_y.P_y$ by N_z cells for a total of $P_x.P_y$ processors. Hence each processor has a sub-domain of N_x by N_y by N_z cells.

The runtime for one cycle of Sweep3D is modeled as:

$$T_{cycle} = \left(2.P_x + 4.P_y - 6\right).\left(T_{comp} + \frac{\left(1+P_{SMP}\right)}{CL}.T_{msg}\right) + N_{sweep}.\left(T_{comp} + \frac{2.\left(1+P_{SMP}\right)}{CL}.T_{msg}\right) \quad \text{(B.1)}$$

where P_x and P_y are the number of processors in the X and Y of the 2-D decomposition respectively, T_{comp} is the time to process a cell block, T_{msg} is the time for a message communication, P_{SMP} is the number of processors in a node, and CL is the number of communication links per node.

The time to process a cell block is modelled as:

$$T_{comp} = N_x.N_y.\frac{N_z}{N_b}.\frac{N_\Omega}{N_a}\frac{N_{flops}}{R_{flops}} \quad \text{(B.2)}$$

where N_{flops} is the number of flops per cell, and R_{flops} is the achieved flop rate on a single processor. The number of sweeps for all eight octants is given by:

$$N_{sweep} = 8.N_b.N_a \quad \text{(B.3)}$$

where the number of blocks in the Z dimension is: $N_b = \frac{N_z}{K_b}$ and the number of angle blocks is $N_a = \frac{N_\Omega}{A_b}$ for N_Ω angles per octant. K_b and A_b are input parameters. T_{msg} is given by:

$$T_{msg} = L_c + \frac{N_{msg}}{B_c} \quad \text{(B.4)}$$

where L_c and B_c is the latency and bandwidth of the communication network, and N_{msg} is the size of a message. Assuming $N_x = N_y$, N_{msg} is given by:

$$N_{msg} = 8.N_x.\frac{N_z}{K_b}.\frac{N_\Omega}{N_a} \quad \text{(B.5)}$$

The values of the main parameters used by the Sweep3D performance model in the analysis in Section 4 are listed in Table B.1. The values for the Alpha system have been previously measured. The values for the Earth Simulator are assumed using the best information currently available. These may be refined once accurate performance information is known such as the actual computation time for a grid of size 8x8x280 cells on a single Earth Simulator processor.

Table B.1. Main parameters used in the Sweep3D performance model.

Parameter		Alpha ES45	Earth Simulator
P_{SMP}	Processors per node	4	8
CL	Communication Links per Node	1	1
N_x N_y N_z	Sub-grid size in cells per processor	12 12 280	8 8 280
N_{Ω}	Number of angles per octant	8	8
N_b N_a	Number of blocks in the Z dimension and in angles	70, 2	$\begin{cases} 70,1 & (\%5) \\ 35,1 & (\%10) \\ 28,1 & (\%15) \\ 28,1 & (\%20) \\ 28,1 & (\%25) \\ 28,1 & (\%30) \end{cases}$
N_{flops}	Number of flops per cell	40	40
R_{flops}	Achieved flop rate on a single processor	350Mflops	$\begin{cases} 400Mflops & (\%5) \\ 800Mflops & (\%10) \\ 1200Mflops & (\%15) \\ 1600Mflops & (\%20) \\ 2000Mflops & (\%25) \\ 2400Mflops & (\%30) \end{cases}$
L_c	Uni-directional MPI communication Latency	$\begin{cases} 5.05\mu s & S < 64 \\ 5.47\mu s & 64 \le S \le 512 \\ 10.3\mu s & S > 512 \end{cases}$	$6\mu s$
B_c	Uni-directional MPI communication Bandwidth	$\begin{cases} 0.0 & S < 64 \\ 78MB/s & 64 \le S \le 512 \\ 294MB/s & S > 512 \end{cases}$	$10GB/s$